CS 495: Introduction to Web Science

Fall 2013

Assignment 7

Onapha Rattanachottiteepakorn

Submission Date: November 13, 2013

**Question 1:**

Using D3, create a graph of the Karate club before and after the split.

* Weight the edges with the data from:
  + http://vlado.fmf.uni-lj.si/pub/networks/data/ucinet/zachary.dat
* Have the transition from before/after the split occur on a mouse click.

In order to create the files to use in D3, it was necessary to convert the data file into json files. To do that, the python scripts ‘convertzachary.py’ and ‘split.py’ were created to convert the data and calculate the nodes’ separation respectively.

To map out the nodes and their edges, I created a set of dictionary with ‘nodes’ and ‘links’. The first 41 lines of the data file were ignored, as they were deemed unnecessary for the conversion. The script was written so that it ignores the edges that have no weight and records all the connecting nodes as ‘sources’, ‘weight’, and ‘target’.

The second script, ‘split.py’, separates the nodes into two groups. Working with the json file created from the first script, the program uses networkx as discussed during class to split the nodes according to the weight between each links. The finished product is saved in karateclub\_split.json and indented for easier reading via pretty printing.

The final product for this question can be viewed in Zachary.html. Majority of the code in this file was heavily based on an example from Mike Bostock. The script initializes the graph with the file karateclub.json. If the user clicks on the page, the script switches the json file and draws a new graph. User can toggle between each data with a click anywhere on the webpage. Figure one shows the graph before the split and Figure 2 shows the nodes after the separation.

D3 is an extremely useful script for data visualization. Not only was it capable of displaying nodes and edges, the script’s variety usage allows programmers create projects such as Color (<http://color.method.ac/>) to illustrate that data representation can be dynamic and vibrant rather than the traditional bar or pie graphs. This JavaScript tool among many improves the web usability and revolutionized the way we represent data.

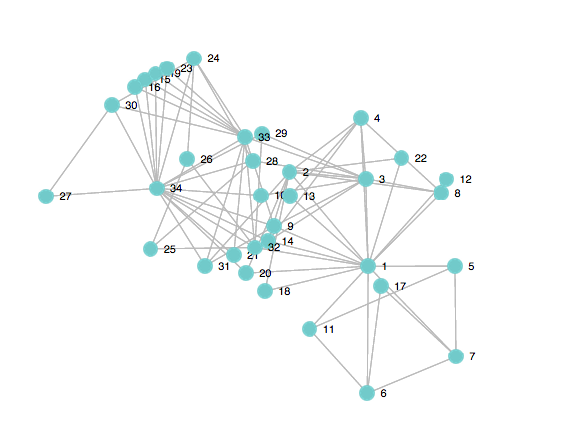
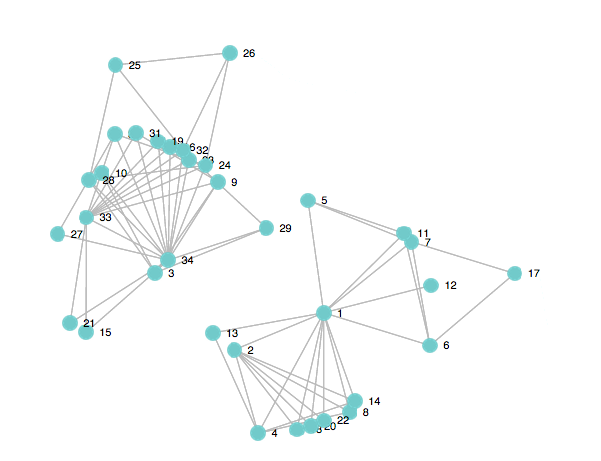


Figure 1: The initial data.

Figure 2: The data after the split.**Sources:**

<http://docs.python.org/2/library/json.html>

<http://stackoverflow.com/questions/16040635/python-writing-things-to-json>

<https://www.dashingd3js.com/creating-svg-elements-based-on-data>